



# Υπολογιστική Λογική και Λογικός Προγραμματισμός

**Ενότητα 10:** Η Γλώσσα Λογικού Προγραμματισμού Prolog  
Εφαρμογές

Σύνθετοι Όροι - Συμβολικά Μαθηματικά: Συμβολική Παραγωγή,  
Φυσικοί Αριθμοί

Νίκος Βασιλειάδης, Αναπλ. Καθηγητής  
Τμήμα Πληροφορικής



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

# Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



# Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης» έχει χρηματοδοτήσει μόνο την αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.





ΑΡΙΣΤΟΤΕΛΕΙΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ  
ΑΚΑΔΗΜΑΪΚΑ  
ΜΑΘΗΜΑΤΑ



**Η Γλώσσα Λογικού Προγραμματισμού**  
**Prolog**  
**Εφαρμογές**  
**Σύνθετοι Όροι - Συμβολικά**  
**Μαθηματικά: Συμβολική Παραγωγή,**  
**Φυσικοί Αριθμοί**

# Συμβολική παραγωγή

| ?- deriv.

Type in the expression to be derivated:

|:  $5*x^4-17*x^3+x^2-5*x+4$ .

Derivate on which variable? |:  $x$ .

Initial expression:

$$5 * x ^ 4 - 17 * x ^ 3 + x ^ 2 - 5 * x + 4$$

Unsimplified derivative:

$$5 * (4 * x ^ 3) + x ^ 4 * 0 - (17 * (3 * x ^ 2) + x ^ 3 * 0) + 2 * x ^ 1 - (5 * 1 + x * 0) + 0$$

Simplified Derivative:

$$5 * (4 * x ^ 3) - 17 * (3 * x ^ 2) + 2 * x - 5$$

yes



# Συμβολική παραγωγή - Απλές Συναρτήσεις

Κανόνας	Αναπαράσταση σε Prolog
Μεταβλητή ως προς τον εαυτό της $d(x)/dx = 1$	<code>der(X,X,1).</code>
Μεταβλητή σε δύναμη $d(x^n)/dx = n*x^{n-1}$	<code>der(X^N,X,N*X^N1) :- number(N), N1 is N - 1.</code>
Σταθερά: $d(c)/dx = 0$	<code>der(C,X,0) :- number(C).</code>
Εκθετική: $d(e^x) / dx = e^x$	<code>der(exp(X),X,exp(X)).</code>
Ημίτονο: $d(\eta\mu(x))/dx = \sigma\upsilon\nu(x)$	<code>der(sin(X),X,cos(X)).</code>
Συνημίτονο: $d(\sigma\upsilon\nu(x))/dx = -\eta\mu(x)$	<code>der(cos(X),X,-sin(X)).</code>
Λογάριθμος: $d(\lambda\omicron\gamma(x))/dx = 1/x$	<code>der(ln(X),X,1/X).</code>



# Συμβολική παραγωγή - Πράξεις Συναρτήσεων (1/2)

Κανόνας	Αναπαράσταση σε Prolog
<p>Άθροισμα Συναρτήσεων</p> $d(f(x)+g(x))/dx = d(f(x))/dx + d(g(x))/dx$	<p><b>der(F+G,X,DF+DG) :-</b> <b>der(F,X,DF), der(G,X,DG).</b></p>
<p>Διαφορά Συναρτήσεων</p> $d(f(x)-g(x))/dx = d(f(x))/dx - d(g(x))/dx$	<p><b>der(F-G,X,DF-DG) :-</b> <b>der(F,X,DF), der(G,X,DG).</b></p>
<p>Αντίθετη Συνάρτησης</p> $d(-f(x))/dx = -d(f(x)) / dx$	<p><b>der(-F,X,-DF) :-</b> <b>der(F,X,DF).</b></p>
<p>Γινόμενο Συναρτήσεων</p> $d(f(x)*g(x))/dx =$ $f(x)*d(g(x))/dx + g(x)*d(f(x))/dx$	<p><b>der(F*G,X,F*DG+DF*G) :-</b> <b>der(F,X,DF), der(G,X,DG).</b></p>



# Συμβολική παραγωγή - Πράξεις Συναρτήσεων (2/2)

Κανόνας	Αναπαράσταση σε Prolog
Αντίστροφη Συνάρτησης $d(1/f(x))/dx = - d(f(x))/dx/f(x)^2$	<b>der(1/F,X,-DF/F^2) :- der(F,X,DF).</b>
Πηλίκο Συναρτήσεων $d(f(x)/g(x))/dx =$ $(g(x)*d(f(x))/dx - f(x)*d(g(x))/dx)/g(x)^2$	<b>der(F/G,X,(G*DF-F*DG)/G^2) :- der(F,X,DF), der(G,X,DG).</b>





# Συμβολική παραγωγή - Σύνθεση Συναρτήσεων

Κανόνας	Αναπαράσταση σε Prolog
Σύνθεση Συναρτήσεων $d(f(g(x)))/dx =$ $d(f(g(x)))/dg(x) * d(g(x))/dx$	<b>der(F_G_X,X,DF*DG) :-</b> <b>F_G_X =.. [F,G_X],</b> <b>der(F_G_X,G_X,DF),</b> <b>der(G_X,X,DG).</b>
Αν μπορούσαμε να έχουμε μεταβλητές στη θέση συναρτησιακών συμβόλων	<b>der(F(G),X,DF*DG) :-</b> <b>der(F(G),G,DF),</b> <b>der(G,X,DG).</b>



# Συμβολική παραγωγή – Παραδείγματα (1/3)

- Συνάρτηση:  $F(x) = \eta\mu(x) + \sigma\upsilon\nu(x)$

?-  $\text{der}(\sin(x)+\cos(x),x,D)$ .

$$D = \cos(x) + (-\sin(x))$$

- Συνάρτηση:  $F(x) = e^x - \log(x)$

?-  $\text{der}(\exp(x) - \ln(x),x,D)$ .

$$D = \exp(x) - 1/x$$

- Συνάρτηση:  $F(x) = 5 * x^2$

?-  $\text{der}(5 * x^2,x,D)$ .

$$D = 5 * (2 * x^1) + x^2 * 0$$



# Συμβολική παραγωγή – Παραδείγματα (2/3)

- Συνάρτηση:  $F(x) = \eta\mu(\sigma\upsilon\nu(x))$

?-  $\text{der}(\sin(\cos(x)),x,D)$ .

$$D = \cos(\cos(x)) * (-\sin(x))$$

- Συνάρτηση:  $F(x) = 1/x$

?-  $\text{der}(1/x,x,D)$ .

$$D = -1 / x^2$$

- Συνάρτηση:  $F(x) = x^{-1}$

?-  $\text{der}(x^{-1},x,D)$ .

$$D = -1 * x^{-2}$$



# Συμβολική παραγωγή – Παραδείγματα (3/3)

- Συνάρτηση:  $F(x) = x^3 * \log(x)$ .

?-  $\text{der}(x^3 * \ln(x), x, D)$ .

$$D = x^3 * (1 / x) + \ln(x) * (3 * x^2)$$

- Συνάρτηση:  $F(x) = \eta\mu(x) / \sigma\upsilon\nu(x)$

?-  $\text{der}(\sin(x)/\cos(x), x, D)$ .

$$D = (\cos(x) * \cos(x) - \sin(x) * (-\sin(x))) / \cos(x)^2$$

- Συνάρτηση: Το ολοκλήρωμα της  $F(x) = -\eta\mu(x) + \sigma\upsilon\nu(x)$ .

?-  $\text{der}(O, x, -\sin(x) + \cos(x))$ .

$$O = \cos(x) + \sin(x)$$



# Απλοποίηση αλγεβρικών εκφράσεων

## Πρόσθεση

**$\text{simplify}(A + 0, A1) :- \text{simplify}(A, A1).$**

**$\text{simplify}(0 + B, B1) :- \text{simplify}(B, B1).$**

**$\text{simplify}(A + B, E) :-$**

**$\text{simplify}(A, A1),$**

**$\text{simplify}(B, B1),$**

**$(A1 \backslash= A; B1 \backslash= B),$**

**$\text{simplify}(A1 + B1, E).$**



# Απλοποίηση αλγεβρικών εκφράσεων

## Αφαίρεση

**simplify(A - 0, A1) :- simplify(A, A1).**

**simplify(0 - B, -B1) :- simplify(B, B1).**

**simplify(A - B, E) :-**  
    **simplify(A, A1),**  
    **simplify(B, B1),**  
    **(A1 \= A; B1 \= B),**  
    **simplify(A1 - B1, E).**



# Απλοποίηση αλγεβρικών εκφράσεων

## Πολλαπλασιασμός

**simplify(A \* 0, 0).**

**simplify(0 \* A, 0).**

**simplify(1 \* A, A1) :- simplify(A, A1).**

**simplify(A \* 1, A1) :- simplify(A, A1).**

**simplify(A \* B, E) :-**

**simplify(A, A1),**

**simplify(B, B1),**

**(A1 \= A; B1 \= B),**

**simplify(A1 \* B1, E).**



# Απλοποίηση αλγεβρικών εκφράσεων

## Διαίρεση

**simplify(0 / A, 0).**

**simplify(A / 1, A1) :- simplify(A, A1).**

**simplify(A / B, E) :-**

**simplify(A, A1),**

**simplify(B, B1),**

**(A1 \= A; B1 \= B),**

**simplify(A1 / B1, E).**





# Απλοποίηση αλγεβρικών εκφράσεων

## Ύψωση σε δύναμη

$\text{simplify}(A^0, 1)$ .

$\text{simplify}(A^1, A1) :- \text{simplify}(A, A1)$ .

$\text{simplify}(0^E, 0)$ .

$\text{simplify}(1^E, 1)$ .

$\text{simplify}(A^B, E) :-$

$\text{simplify}(A, A1),$

$\text{simplify}(B, B1),$

$(A1 \backslash= A; B1 \backslash= B),$

$\text{simplify}(A1^B1, E)$ .



# Απλοποίηση αλγεβρικών εκφράσεων

- Διπλή αντίθεση

**$\text{simplify}(\text{--}(\text{--}(A)), A1) :- \text{simplify}(A, A1).$**

- Αν κανένας κανόνας απλοποίησης δεν ισχύει, τότε επιστρέφεται η αλγεβρική έκφραση όπως ήταν

**$\text{simplify}(A, A).$**



# Ολοκληρωμένο Πρόγραμμα

**deriv :-**

```
write('Type in the expression to be derivated: '), nl,  
read(Expression), nl,  
write('Derivate on which variable? '),  
read(Var), nl,  
der(Expression,Var,Derivative),  
simplify(Derivative,SimpDer),  
write('Initial expression:'), nl,  
write(Expression), nl,  
write('Unsimplified derivative:'), nl,  
write(Derivative), nl,  
write('Simplified Derivative:'), nl,  
write(SimpDer), nl.
```



# Εκτέλεση Προγράμματος

| ?- deriv.

Type in the expression to be derivated:

|:  $5*x^4-17*x^3+x^2-5*x+4$ .

Derivate on which variable? |:  $x$ .

Initial expression:

$$5 * x ^ 4 - 17 * x ^ 3 + x ^ 2 - 5 * x + 4$$

Unsimplified derivative:

$$5 * (4 * x ^ 3) + x ^ 4 * 0 - (17 * (3 * x ^ 2) + x ^ 3 * 0) + 2 * x ^ 1 - (5 * 1 + x * 0) + 0$$

Simplified Derivative:

$$5 * (4 * x ^ 3) - 17 * (3 * x ^ 2) + 2 * x - 5$$

yes



# Απλοποίηση αλγεβρικών εκφράσεων

## - Ειδική Περίπτωση

- Πολλαπλασιασμός αριθμού με παρένθεση στην οποία υπάρχει γινόμενο και το πρώτο μέλος του γινομένου είναι επίσης αριθμός!

**simplify(A \* (B \* C), D \* C) :-**

**number(A),**

**number(B),**

**D is A \* B.**



# Εκτέλεση προγράμματος - 2

| ?- deriv.

Type in the expression to be derivated:

|:  $5*x^4-17*x^3+x^2-5*x+4$ .

Derivate on which variable? |:  $x$ .

Initial expression:

$$5 * x ^ 4 - 17 * x ^ 3 + x ^ 2 - 5 * x + 4$$

Unsimplified derivative:

$$5 * (4 * x ^ 3) + x ^ 4 * 0 - (17 * (3 * x ^ 2) + x ^ 3 * 0) + 2 * x ^ 1 - (5 * 1 + x * 0) + 0$$

Simplified Derivative:

$$20 * x ^ 3 - 51 * x ^ 2 + 2 * x - 5$$

yes



# Φυσικοί Αριθμοί ως Σύνθετοι Όροι

- Ένας φυσικός αριθμός μπορεί να αναπαρασταθεί ως σύνθετος όρος ως εξής:
  - Το μηδέν (0) αναπαρίσταται ως απλός όρος: **0**
  - Το ένα (1) αναπαρίσταται ως ο σύνθετος όρος: **s(0)**
    - Σημασία: Το 1 είναι ο επόμενος αριθμός από το μηδέν.
  - Το δύο (2) αναπαρίσταται ως ο σύνθετος όρος: **s(s(0))**
  - ...
- Να γραφεί κατηγορημα **natural\_number/1** το οποίο να πετυχαίνει όταν το όρισμά του είναι φυσικός αριθμός.

**natural\_number(0).**

**natural\_number(s(X)) :-**

**natural\_number(X).**



# Κατηγορημα natural\_number/1

- Το κατηγορημα ελέγχει αν κάποιος αριθμός είναι φυσικός, αλλά και μπορεί να παράγει όλους τους φυσικούς αριθμούς.

?- natural\_number(0).

yes

?- natural\_number(s(s(0))).      % 2

yes

?- natural\_number(N).

N = 0 ;

N = s(0) ;

N = s(s(0)) ;

...





# Εκτέλεση natural\_number/1 (1/2)

natural\_number(o).

natural\_number(s(X)) :-  
natural\_number(X).

?- natural\_number(s(s(o))).

| {X=s(o)}

?- natural\_number(s(o)).

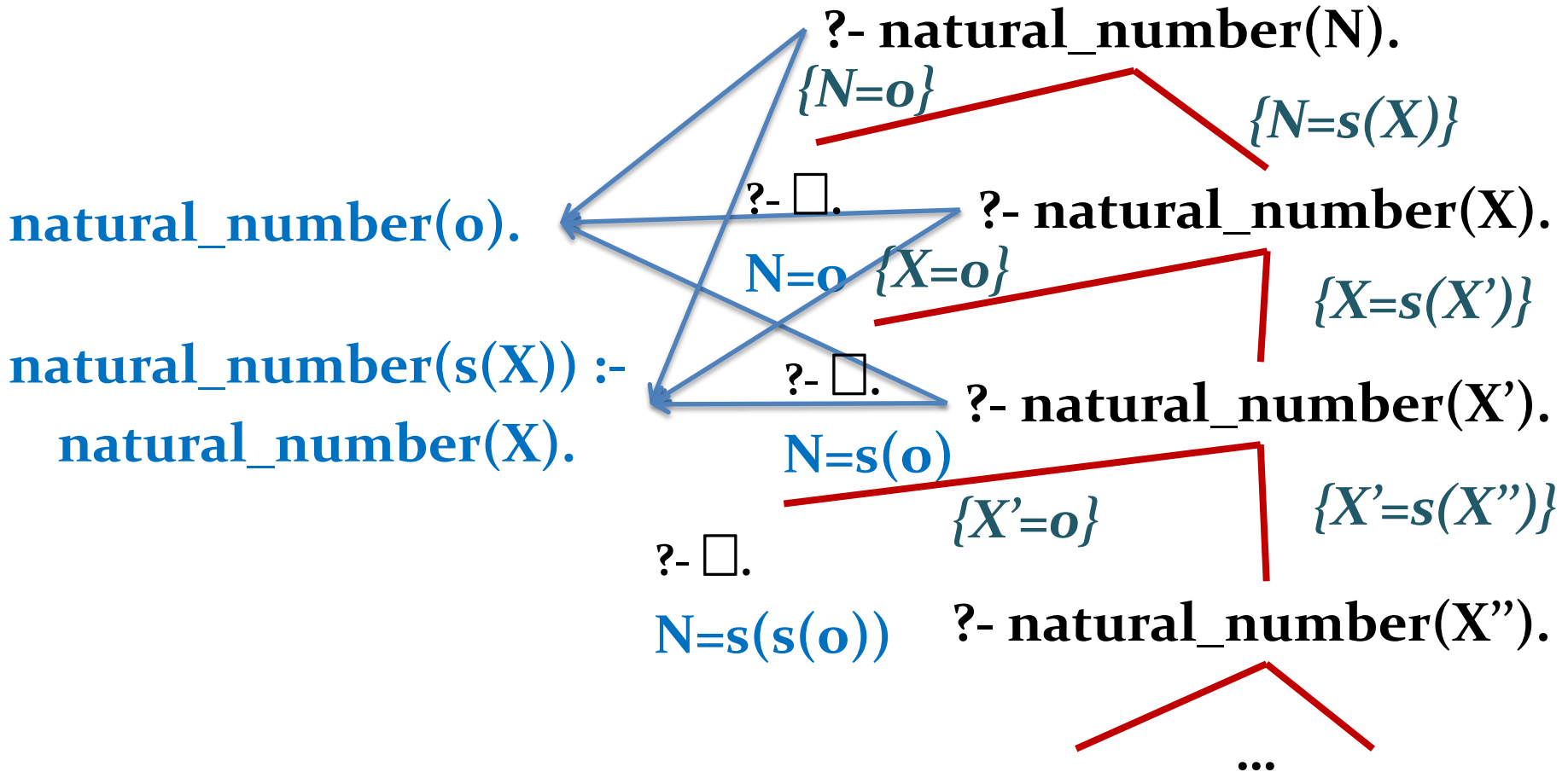
| {X'=o}

?- natural\_number(o).

?- □.



# Εκτέλεση natural\_number/1 (2/2)



# Φυσικοί Αριθμοί – Πρόσθεση (1/3)

- Να γραφεί κατηγορημα **plus/3** το οποίο να προσθέτει δύο φυσικούς αριθμούς και να επιστρέφει το άθροισμά τους στο τρίτο όρισμα.

**plus(0,X,X).**

*% Μηδέν και X μας δίνει X*

**plus(s(X),Y,s(Z)) :-** *% Για να προσθέσω (X+1) και Y*

**plus(X,Y,Z).**

*% αρκεί να προσθέσω X και Y*

*% και στο αποτέλεσμα να προσθέσω*

*1*



# Φυσικοί Αριθμοί – Πρόσθεση (2/3)

- Το παραπάνω κατηγορημα έχει πολλαπλές λειτουργίες:
  - Ελέγχει αν κάποιος αριθμός είναι το άθροισμα άλλων δύο.

?- plus(s(0),s(s(0)),s(s(s(0))))). % 1+2=3

yes

?- plus(s(0),s(s(0)),s(s(s(s(0))))). % 1+2=4

no

- Υπολογίζει το άθροισμα δύο φυσικών αριθμών.

?- plus(s(0),s(s(0)),C). % 1+2=?

C = s(s(s(0)))



# Φυσικοί Αριθμοί – Πρόσθεση (3/3)

- Το παραπάνω κατηγορημα έχει πολλαπλές λειτουργίες:
  - Υπολογίζει τη διαφορά δύο φυσικών αριθμών!

?- plus(s(0),B,s(s(0))). *% 1+?=3*

**B = s(s(0))**

?- plus(A,s(s(0)),s(s(s(0)))). *% ?+2=3*

**A = s(0)**

- Βρίσκει όλα τα ζεύγη αριθμών που αν προστεθούν μπορούν να μας δώσουν έναν φυσικό αριθμό.

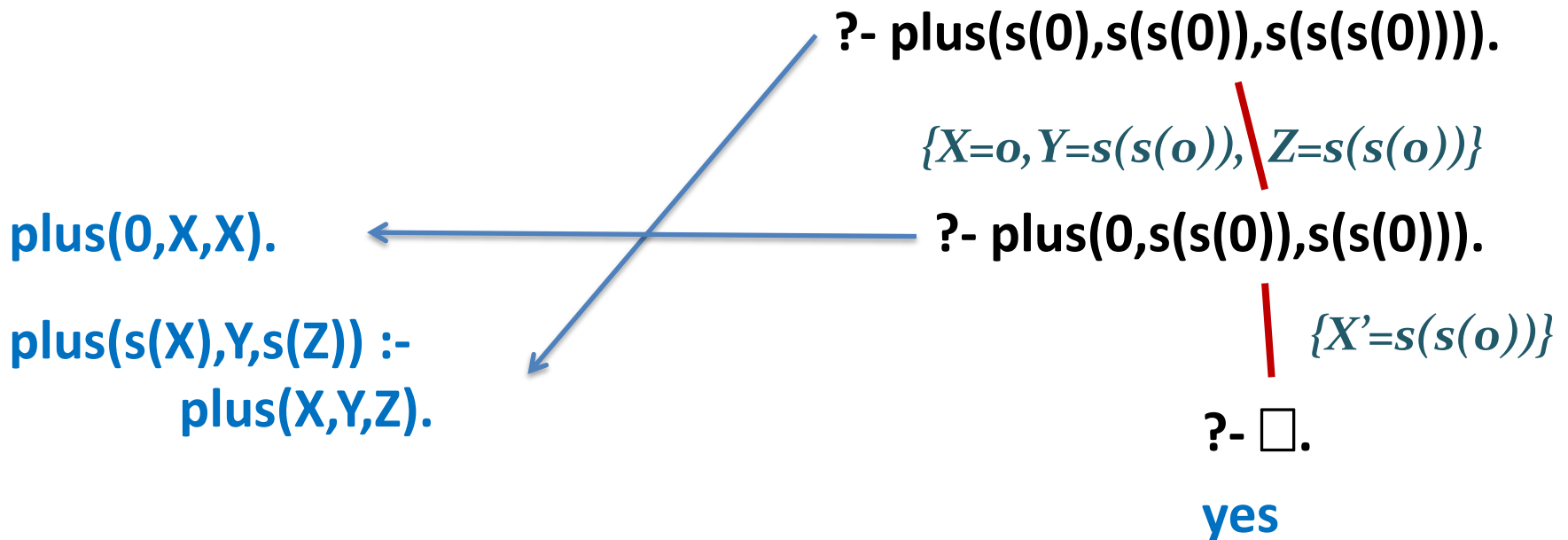
?- plus(A,B,s(s(s(0)))). *% ?+?=3*

**A = 0 , B = s(s(s(0))) ;      A = s(0) , B = s(s(0)) ;**

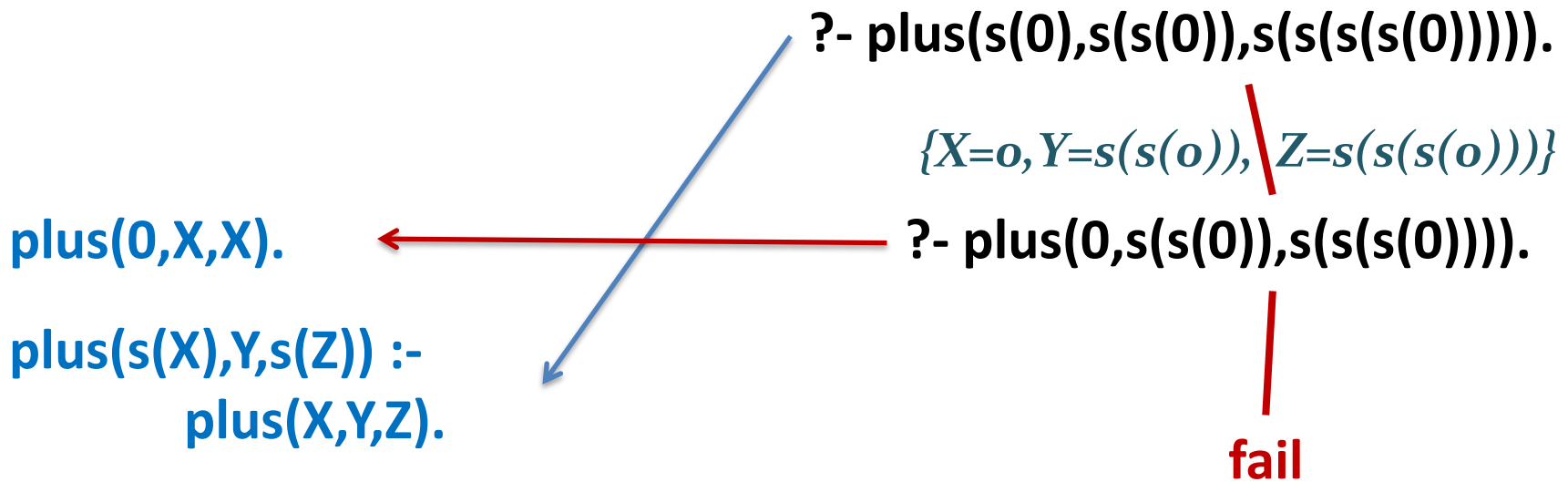
**A = s(s(0)) , B = s(0) ;      A = s(s(s(0))) , B = 0 ;      no**



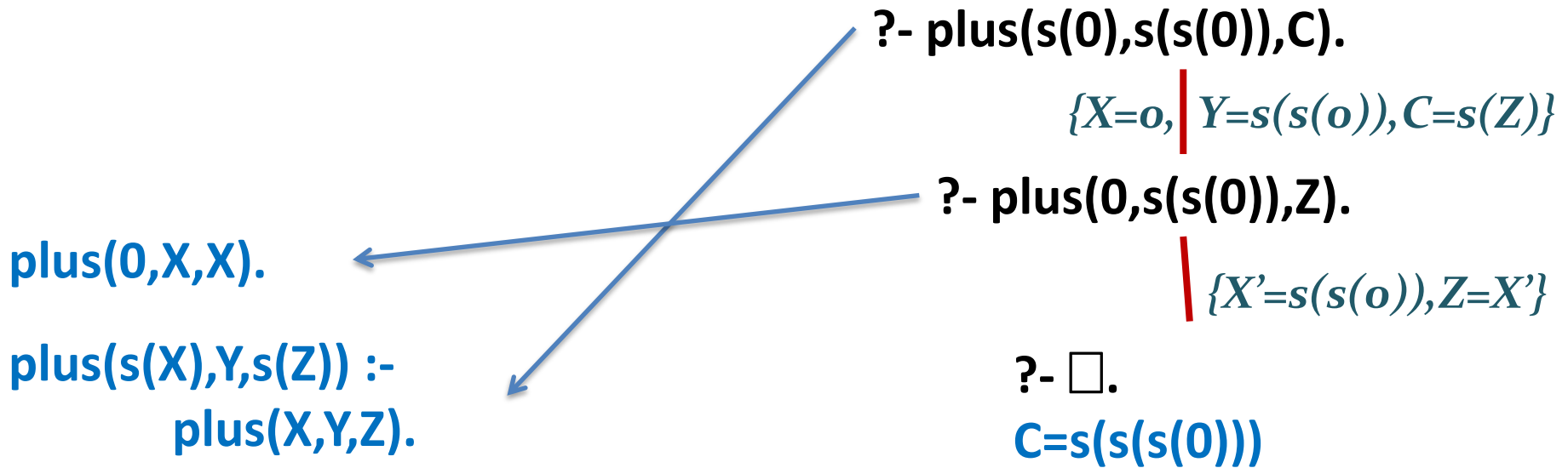
# Έλεγχος αν κάποιος αριθμός είναι το άθροισμα άλλων δύο (1/2)



# Έλεγχος αν κάποιος αριθμός είναι το άθροισμα άλλων δύο (2/2)



# Υπολογισμός αθροίσματος δύο φυσικών αριθμών





# Υπολογισμός διαφοράς δύο φυσικών αριθμών (1/2)

$\text{plus}(0, X, X).$

$\text{plus}(s(X), Y, s(Z)) :-$   
 $\text{plus}(X, Y, Z).$

$?- \text{plus}(s(0), B, s(s(s(0))))).$

$\{X=0, B=Y, Z=s(s(0))\}$

$?- \text{plus}(0, B, s(s(0))).$

$\{B=X'=s(s(0))\}$

$?- \square.$

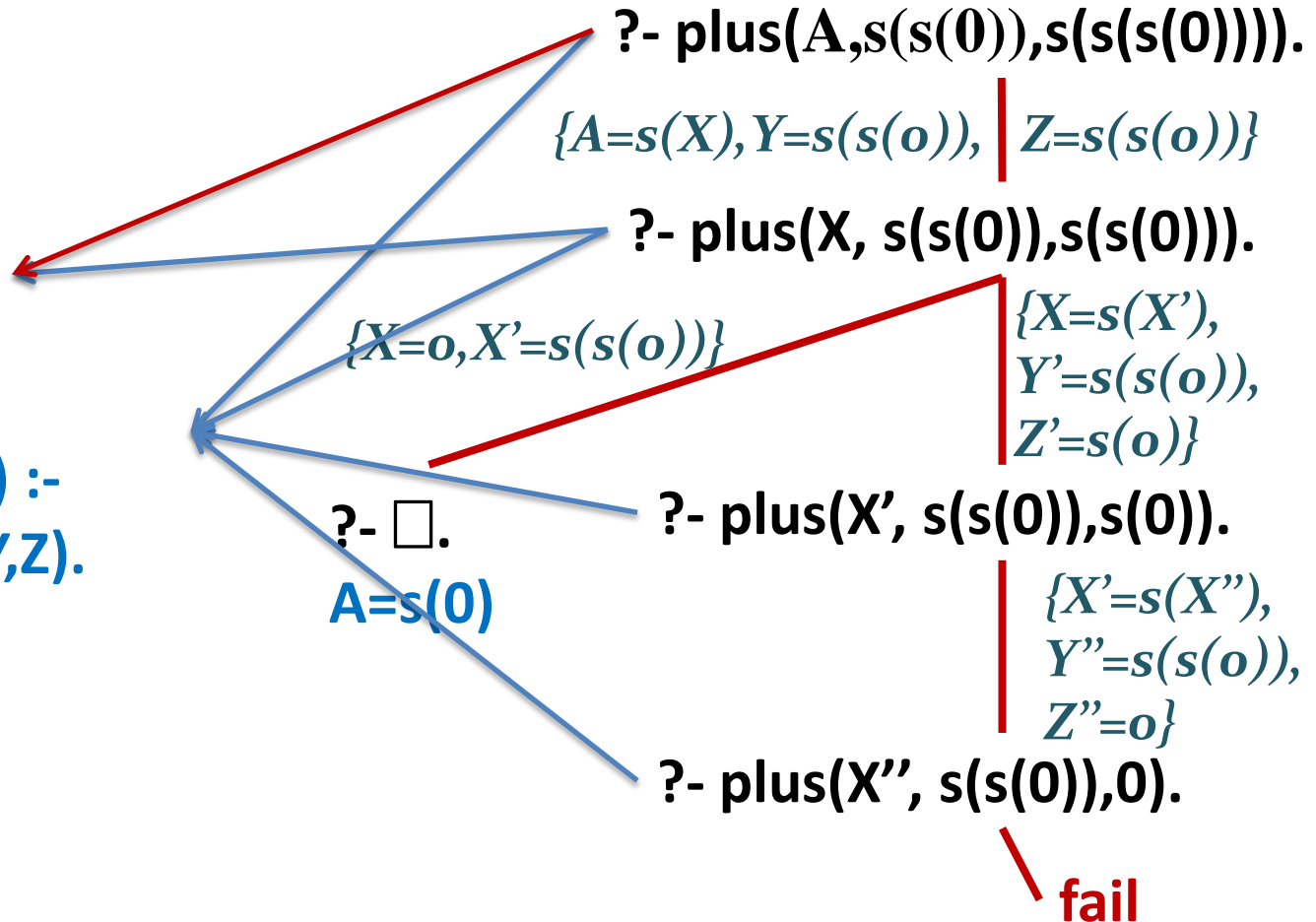
$B=s(s(0))$



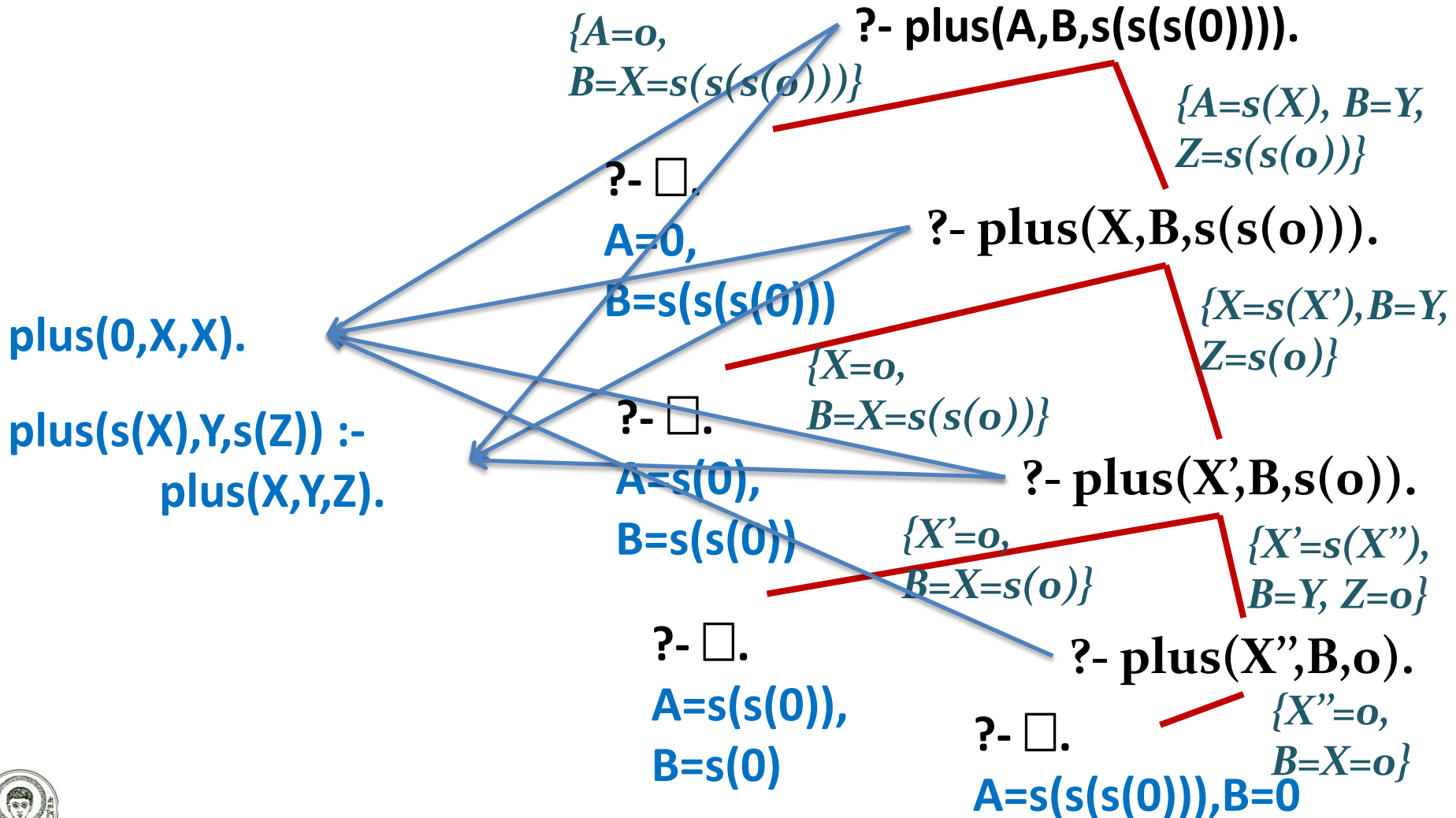
# Υπολογισμός διαφοράς δύο φυσικών αριθμών (2/2)

$\text{plus}(0, X, X).$

$\text{plus}(s(X), Y, s(Z)) :-$   
 $\text{plus}(X, Y, Z).$



# Ζεύγη Προσθετών Φυσικού Αριθμού



# Φυσικοί Αριθμοί – Πολλαπλασιασμός (1/3)

- Να γραφεί κατηγορημα **times/3** το οποίο να πολλαπλασιάζει δύο φυσικούς αριθμούς και να επιστρέφει το γινόμενό τους στο τρίτο όρισμα.
  - Εκμεταλλευόμαστε την ύπαρξη της άθροισης.

**times(0,Y,0).**                      *% Το 0 απορροφητικό  
στοιχείο*

**times(s(X),Y,Z1) :-**            *%  $(X+1)*Y = (X*Y) + Y$*

**times(X,Y,Z),**

**plus(Z,Y,Z1).**



# Φυσικοί Αριθμοί – Πολλαπλασιασμός (2/3)

- Το παραπάνω κατηγορημα έχει πολλαπλές λειτουργίες
  - Πολλαπλασιασμός 2 αριθμών (και έλεγχος).

?- `times(s(s(0)),s(s(s(0))),C)`. %  $2*3=?$

`C = s(s(s(s(s(s(0))))))` % 6

- Διαίρεση!

?- `times(s(s(0)),B,s(s(s(s(s(s(0))))))`. %  $2*?=6$

`B = s(s(s(0)))` % 3

?- `times(A,s(s(0)),s(s(s(s(s(s(0))))))`. %  $?*2=6$

`A = s(s(s(0)))` % 3



# Φυσικοί Αριθμοί – Πολλαπλασιασμός (3/3)

- Το παραπάνω κατηγορημα έχει πολλαπλές λειτουργίες:
  - Ανάλυση αριθμού σε γινόμενο δύο άλλων.

?-  $\text{times}(A, B, s(s(s(s(s(0))))))$ .

%  $? * ? = 6$

$A = s(0)$  ,  $B = s(s(s(s(s(0)))))$  ;

%  $1 * 6 = 6$

$A = s(s(0))$  ,  $B = s(s(s(0)))$  ;

%  $2 * 3 = 6$

$A = s(s(s(0)))$  ,  $B = s(s(0))$  ;

%  $3 * 2 = 6$

$A = s(s(s(s(s(s(0))))))$  ,  $B = s(0)$

%  $6 * 1 = 6$



# Πολλαπλασιασμός δύο αριθμών

$\text{times}(0, Y, 0).$

$\text{times}(s(X), Y, Z1) :-$   
 $\text{times}(X, Y, Z),$   
 $\text{plus}(Z, Y, Z1).$

?-  $\text{times}(s(s(o)), s(s(s(o))), C).$

$\{X=s(o),$   
 $Y=s(s(s(o))), C=Z1\}$

?-  $\text{times}(s(o), s(s(s(o))), Z),$   
 $\text{plus}(Z, s(s(s(o))), C).$

$\{X'=o, Y=s(s(s(o))),$   
 $Z=Z1'\}$

?-  $\text{times}(o, s(s(s(o))), Z'),$   
 $\text{plus}(Z', s(s(s(o))), Z),$   
 $\text{plus}(Z, s(s(s(o))), C).$

$\{Y=s(s(s(o))), Z'=o\}$

?-  $\text{plus}(o, s(s(s(o))), Z),$   
 $\text{plus}(Z, s(s(s(o))), C).$

$\{Z=s(s(s(o))),$   
 $C=s(s(s(s(s(s(o))))))\} \dots$



# Φυσικοί Αριθμοί – Ύψωση σε δύναμη (1/3)

- Να γραφεί κατηγορημα **power/3** το οποίο να υψώνει έναν αριθμό σε μία δύναμη (επίσης φυσικός αριθμός) και να επιστρέφει το αποτέλεσμα στο τρίτο όρισμα.
  - Εκμεταλλευόμαστε την ύπαρξη του πολλαπλασιασμού.

**power(X,0,s(0)).**                      %  $X^0 = 1$

**power(X,s(Y),Z1) :-** %  $X^{Y+1} = X^Y \bullet X$

**power(X,Y,Z),**

**times(Z,X,Z1).**





# Φυσικοί Αριθμοί – Ύψωση σε δύναμη (2/3)

- Το παραπάνω κατηγορημα έχει πολλαπλές λειτουργίες.
  - Ύψωση σε δύναμη (και έλεγχος).

?- `power(s(s(0)),s(s(0)),C).`

$$\% 2^2=4$$

`C = s(s(s(s(0))))`

$$\sqrt[3]{8} = 2$$

- Εύρεση Ρίζας!

?- `power(A,s(s(s(0))),s(s(s(s(s(s(s(0))))))))).`

`A = s(s(0))`



# Φυσικοί Αριθμοί – Ύψωση σε δύναμη (3/3)

- Το παραπάνω κατηγορημα έχει πολλαπλές λειτουργίες.
  - **Λογάριθμος**: Σε ποια δύναμη αν υψωθεί κάποιος αριθμός μας δίνει κάποιον άλλο?

?-  $\text{power}(s(s(s(0))), B, s(s(s(s(s(s(s(s(0))))))))))$ .

$B = s(s(0))$

$$3^B = 9 \Leftrightarrow \log_3(9) = B$$

- Ποιος αριθμός αν υψωθεί σε ποια δύναμη μας δίνει κάποιον άλλο?

?-  $\text{power}(A, B, s(s(s(s(s(s(s(s(0))))))))))$ .

$$A^B = 9$$

$A = s(s(s(s(s(s(s(s(0))))))))), B = s(0);$

$$\% 9^1=9$$

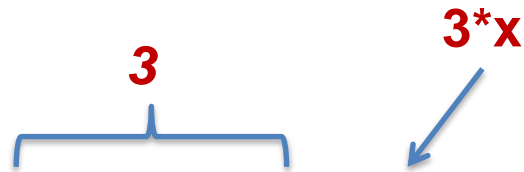
$A = s(s(s(0))), B = s(s(0))$

$$\% 3^2=9$$



# Επίλυση Εξισώσεων (1/2)

- $3 * x + 5 = 11$



?- times(s(s(s(0))),X,Y),

plus(Y,s(s(s(s(s(0))))),s(s(s(s(s(s(s(s(s(s(0))))))))))

X = s(s(0)),



Y = s(s(s(s(s(s(0))))))



# Επίλυση Εξισώσεων (2/2)

- Γενικώς:  $A * x + B = C$

**equation(A,B,C,X) :-**

**times(A,X,Y), plus(Y,B,C).**

?- equation( $\overbrace{s(s(s(0)))}^3$ ,  $\overbrace{s(s(s(s(s(0))))}^5$ ),  
 $\underbrace{s(s(s(s(s(s(s(s(s(s(0))))))))}_{11}$ ), X).

X =  $s(s(0))$  ←  $\underbrace{\hspace{10em}}_2$



# Σύστημα Εξισώσεων (1/2)

- Σύστημα 2 εξισώσεων με 2 αγνώστους.

$$\% A1 * X + B1 * Y = C1$$

$$\% A2 * X + B2 * Y = C2$$

`system(equation(A1,B1,C1),equation(A2,B2,C2),X,Y)`

`:-`

`times(A1,X,AX1), times(B1,Y,BY1),`

`plus(AX1,BY1,C1),`

`times(A2,X,AX2), times(B2,Y,BY2),`

`plus(AX2,BY2,C2).`



# Σύστημα Εξισώσεων (2/2)

- Παράδειγμα:

$$- X + 2 * Y = 14$$

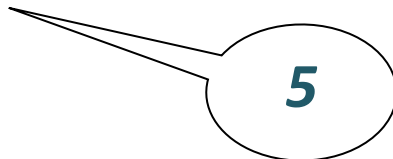
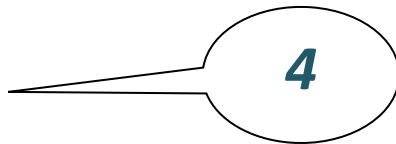
$$- X + Y = 9$$

?- system(equation(s(0),s(s(0)),s(s(...(s(0))))))))),  
 equation(s(0),s(0),s(s(...(s(0)))))),  
 X, Y).



$$X = s(s(s(s(0))))$$

$$Y = s(s(s(s(s(0)))))$$



# Μετατροπή «αραβικών» αριθμών σε σύνθετους όρους

- Απαραίτητο ώστε να είναι «χρήσιμα» όλα τα προηγούμενα κατηγορήματα.

?- `convert(A,11).`

`A = s(s(s(s(s(s(s(s(s(s(0))))))))))`

?- `convert(s(s(s(s(s(s(s(s(s(s(s(0)))))))))),A).`

`A = 11`



# Μετατροπή ψηφίων

`convert_digit(0,0).`

`convert_digit(s(0),1).`

`convert_digit(s(s(0)),2).`

`convert_digit(s(s(s(0))),3).`

`convert_digit(s(s(s(s(0))))),4).`

`convert_digit(s(s(s(s(s(0)))))),5).`

`convert_digit(s(s(s(s(s(s(0))))))),6).`

`convert_digit(s(s(s(s(s(s(s(0))))))))) ,7).`

`convert_digit(s(s(s(s(s(s(s(s(0))))))))),8).`

`convert_digit(s(s(s(s(s(s(s(s(s(0)))))))))) ,9).`

*Δεν γίνεται  
πιο εύκολα*





# Μετατροπή μονοψήφιων αριθμών

**convert(S,N) :-**

**nonvar(N),**

**N < 10,**

**convert\_digit(S,N).**

**convert(S,N) :-**

**var(N),**

**convert\_digit(SN,9),**

**greater\_than\_or\_equal(SN,S),     % 9>=N**

**convert\_digit(S,N).**



# Άλλα χρήσιμα κατηγορήματα

- $X \geq Y$

**greater\_than\_or\_equal(0,0).**

**greater\_than\_or\_equal(s(X),0).**

**greater\_than\_or\_equal(s(X),s(Y)) :-  
greater\_than\_or\_equal(X,Y).**

*Αν θέλουμε  $X > Y$ ,  
τότε αυτή η πρόταση  
πρέπει να φύγει.*

?- **greater\_than\_or\_equal(s(s(0)),s(0)).**

**yes**



# Μετατροπή διψήφιων αριθμών Από αραβικό αριθμό σε σύνθετο όρο

**convert(S,N) :-**

**nonvar(N), N > 9, N < 100,**

**D is N div 10,      % Βρες δεκάδες.**

**U is N mod 10,    % Βρες μονάδες.**

**convert\_digit(SD,D),      % μετέτρεψε τα ψηφία.**

**convert\_digit(SU,U),**

**convert\_digit(SN,9),**

**times(SD,s(SN),SSD),      % Πολ/σε τις δεκάδες με 10.**

**plus(SSD,SU,S).      % Πρόσθεσε τις μονάδες.**



# Μετατροπή διψήφιων αριθμών Από σύνθετο όρο σε αραβικό αριθμό

`convert(S,N) :-`

`var(N),`

`convert_digit(SN,9),`

`greater_than_or_equal(S,s(SN)),`

`my_div(S,s(SN),SSD),`

*% Βρες δεκάδες.*

`my_mod(S,s(SN),SU),`

*% Βρες μονάδες.*

`convert_digit(SSD,D),`

*% μετέτρεψε τα ψηφία.*

`convert_digit(SU,U),`

`N is D * 10 + U.`

*% Πολ/σε τις δεκάδες με 10.*

*% και πρόσθεσε τις μονάδες.*



# Άλλα χρήσιμα κατηγορήματα (1/2)

- Ακέραια διαίρεση.
  - Συνεχής αφαίρεση του  $Y$  από το  $X$ .
  - Μέτρα πόσες αφαιρέσεις γίνονται.
  - Μέχρις ότου το  $X$  γίνει μικρότερο από το  $Y$ .

**my\_div(X,Y,s(Div)) :-**

**greater\_than\_or\_equal(X,Y), !,**

**plus(Y,NextX,X),**

**my\_div(NextX,Y,Div).**

**my\_div(X,Y,0).**

```
?- my_div(s(s(s(s(s(0)))))),s(s(0)),A).  
A = s(s(0)).
```



# Άλλα χρήσιμα κατηγορήματα (2/2)

- Υπόλοιπο ακέραιας διαίρεσης.
  - Συνεχής αφαίρεση του  $Y$  από το  $X$ .
  - Μέχρις ότου το  $X$  γίνει μικρότερο από το  $Y$ .
  - Επέστρεψε την τρέχουσα τιμή του  $X$ .

**my\_mod(X,Y,Mod) :-**

**greater\_than\_or\_equal(X,Y), !,**

**plus(Y,NextX,X),**

**my\_mod(NextX,Y,Mod).**

**my\_mod(X,Y,X).**

```
?- my_mod(s(s(s(s(s(0)))))),s(s(0)),A).  
A = s(0).
```



# Παράδειγμα Συστήματος Εξισώσεων

- $X + 2 * Y = 14$

- $X + Y = 9$

?- `convert(A,14), convert(B,9),  
system( equation(s(0),s(s(0)),A),  
equation(s(0),s(0),B), X, Y),  
convert(X,X1), convert(Y,Y1).`

$$A = s(s(s(s(s(s(s(s(s(...))))))))),$$

$$B = s(s(s(s(s(s(s(s(0))))))))),$$

$$X = s(s(s(s(0))), Y = s(s(s(s(0))))) ,$$

$$X1 = 4, Y1 = 5 .$$



# Μπορούμε να ξεφορτωθούμε τα μαθηματικά της Prolog; (1/2)

```
convert(S,N) :-  
    nonvar(N),  
    N > 9, N < 100,  
    D is N div 10, U is N mod 10,  
    convert_digit(SD,D),  
    convert_digit(SU,U),  
    convert_digit(SN,9),  
    times(SD,s(SN),SSD),  
    plus(SSD,SU,S).
```





# Μπορούμε να ξεφορτωθούμε τα μαθηματικά της Prolog; (2/2)

- Για να γίνει αυτό θα πρέπει να χειριστούμε έναν αριθμό ως συμβολοσειρά.
- Πρόσβαση σε κάθε ψηφίο του αριθμού ξεχωριστά.
  - Λίστα ψηφίων.
- Χρήση ενσωματωμένων κατηγορημάτων της SWI-Prolog.
  - **number\_chars(N,L)**: Μετατρέπει αριθμό σε λίστα χαρακτήρων.
  - **atom\_number(A,N)**: Μετατρέπει ένα άτομο - χαρακτήρα στον αντίστοιχο αριθμό.



# Μετατροπή αριθμού σε λίστα ψηφίων (1/2)

**list\_of\_digits(N,L) :-**

```
nonvar(N),           % Μετατροπή αριθμού σε λίστα.  
number_chars(N,L1),  
chars_to_numbers(L1,L).
```

**list\_of\_digits(N,L) :-**

```
var(N),             % Μετατροπή λίστας σε αριθμό.  
chars_to_numbers(L1,L),  
number_chars(N,L1).
```

**chars\_to\_numbers([],[]).**

**chars\_to\_numbers([H|T],[H1|T1]) :-**

```
atom_number(H,H1),  
chars_to_numbers(T,T1).
```



# Μετατροπή αριθμού σε λίστα ψηφίων (2/2)

?- list\_of\_digits(15,A).

A = [1, 5]

?- list\_of\_digits(B,[1,5]).

B = 15



# NEA Μετατροπή μονοψήφιων αριθμών

**convert(S,N) :-**

**nonvar(N),**

~~**N < 10,**~~      *% Δεν χρειάζεται γιατί το convert\_digit*

**convert\_digit(S,N).**      *% αποτυγχάνει για N < 10.*

**convert(S,N) :-**

**var(N),**

**convert\_digit(SN,9),**

**greater\_than\_or\_equal(SN,S),**

**convert\_digit(S,N).**



# NEA Μετατροπή διψήφιων αριθμών Από αραβικό αριθμό σε σύνθετο όρο

convert(S,N) :-

nonvar(N),

~~N > 9, N < 100, D is N div 10, U is N mod 10,~~

list\_of\_digits(N,[D,U]), % Βρες δεκάδες-μονάδες.

convert\_digit(SD,D), % μετέτρεψε τα ψηφία.

convert\_digit(SU,U),

convert\_digit(SN,9),

times(SD,s(SN),SSD), % Πολ/σε τις δεκάδες με 10.

plus(SSD,SU,S). % Πρόσθεσε τις μονάδες.



# NEA Μετατροπή διψήφιων αριθμών Από σύνθετο όρο σε αραβικό αριθμό

convert(S,N) :-

var(N),

convert\_digit(SN,9),

greater\_than\_or\_equal(S,s(SN)),

my\_div(S,s(SN),SSD),

*% Βρες δεκάδες.*

my\_mod(S,s(SN),SU),

*% Βρες μονάδες.*

convert\_digit(SSD,D),

*% μετέτρεψε τα ψηφία.*

convert\_digit(SU,U),

~~**$N$  is  $D * 10 + U$ .**~~

*% Πολ/σε τις δεκάδες με 10.*

list\_of\_digits(N,[D,U]). *% και πρόσθεσε τις μονάδες.*



# Σημείωμα Αναφοράς

Copyright Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, Νίκος Βασιλειάδης.  
«Υπολογιστική Λογική και Λογικός Προγραμματισμός. Η Γλώσσα Λογικού  
Προγραμματισμού Prolog. Εφαρμογές. Σύνθετοι Όροι - Συμβολικά  
Μαθηματικά: Συμβολική Παραγωγή, Φυσικοί Αριθμοί». Έκδοση: 1.0.  
Θεσσαλονίκη 2014. Διαθέσιμο από τη δικτυακή διεύθυνση:  
<http://eclass.auth.gr/courses/OCRS163/>



# Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

[1] <http://creativecommons.org/licenses/by-nc-sa/4.0/>







# Τέλος ενότητας

Επεξεργασία: Εμμανουήλ Ρήγας  
Θεσσαλονίκη, Εαρινό Εξάμηνο 2013-2014



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ  
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ  
*επένδυση στην κοινωνία της γνώσης*

ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



ΑΡΙΣΤΟΤΕΛΕΙΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΘΕΣΣΑΛΟΝΙΚΗΣ

---

# Σημειώματα

# Διατήρηση Σημειωμάτων

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων
- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.

